IN THE CLAIMS

Please amend the claims as follows:

1. (Currently Amended) A method of handling register spills in a CPU having parallel registers ~~parallel register architecture~~, comprising:

   (i) determining ~~whether~~ that register spill instructions in spill code generated by a register allocator can be associated with each other;

   (ii) based on the determining, ~~if said register spill instructions can be associated, then~~ rewriting said register spill instructions as a parallel register spill instruction; and

   (iii) based on said rewritten parallel register spill instruction, configuring storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel.

2. (Currently Amended) The method of claim 1, wherein said CPU having parallel registers ~~parallel architecture comprises~~ includes a ~~primary~~ first register set and a ~~secondary~~ second register set, and wherein (i) ~~comprises~~ includes determining ~~whether~~ that two register spill instructions can be paired.

3. (Currently Amended) The method of claim [[2]] 1, wherein (i) ~~further comprises~~ includes determining ~~whether~~ that said ~~two~~ register spill instructions are in a basic block within said spill code.

4. (Currently Amended) The method of claim [[3]] 1, wherein (i) ~~further comprises~~ includes determining ~~whether~~ that said ~~two~~ register spill instructions relate to matching register locations in each of ~~said primary~~ a first register set and ~~said secondary~~ a second register set.

5. (Currently Amended) The method of claim [[4]] 1, wherein (i) ~~further comprises~~ includes determining ~~whether any~~ that no intervening instructions between said register spill instructions modify ~~either~~ any of said register spill instructions.

6. (Currently Amended) The method of claim 2, wherein (iii) ~~comprises~~ includes first allocating space on a memory stack to all paired register spills, then allocating space on said memory stack for any remaining register spills.

7. (Currently Amended) The method of claim 2, wherein (iii) ~~comprises~~ includes allocating space on ~~said~~ a memory stack such that paired register spills are double word aligned.

8. (Currently Amended) The method of claim 7, further comprising loading said paired register spill instructions from said memory stack back into matching register locations in each of said ~~primary~~ first register set and said ~~secondary~~ second register set in parallel.

9. (Currently Amended) A system for handling register spills in a CPU having parallel registers ~~parallel register architecture~~, said system comprising:

(a) [[a]] an analyzer module ~~for analyzing~~ configured to analyze spill code generated by a register allocator to determine ~~whether~~ that register spill instructions can be associated with each other;

(b) a rewriter module ~~for rewriting~~ configured to, based on the determining, rewrite said register spill instructions as a parallel register spill instruction~~, if said register spill instructions can be associated~~; and

(c) a storage module ~~for configuring~~ configured to configure storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel based on said rewritten parallel register spill instruction.

10. (Currently Amended) The system of claim 9, wherein said CPU having parallel registers ~~parallel architecture comprises~~ includes a ~~primary~~ first register set and a ~~secondary~~ second register set, and said analyzer module in (a) is configured to determine ~~whether~~ that two register spill instructions can be paired.

11. (Currently Amended) The system of claim [[10]] 9, wherein said analyzer module in (a) is ~~further~~ configured to determine ~~whether~~ that said ~~two~~ register spill instructions relate to matching register locations in each of ~~said primary~~ a first register set and ~~said secondary~~ a second register set.

12. (Currently Amended) The system of claim [[11]] 9, wherein said analyzer module in (a) is ~~further~~ configured to determine ~~whether any~~ that no intervening instructions between said register spill instructions modify ~~either~~ any of said register spill instructions.

13. (Currently Amended) The system of claim [[12]] 10, wherein said storage module in (c) is configured to first allocate space on a memory stack to all paired register spills, then allocate space on said memory stack for any remaining register spills.

14. (Currently Amended) The system of claim [[12]] 10, wherein said storage module in (c) is configured to allocate space on ~~said~~ a memory stack such that paired register spills are double word aligned.

15. (Currently Amended) The system of claim 14, further comprising loading said paired register spills from said memory stack back into matching register locations in each of said ~~primary~~ first register set and said ~~secondary~~ second register set in parallel.

16. (Currently Amended) A system for handling register spills in a CPU having parallel registers ~~parallel register architecture~~, said system comprising:

(a) means for determining that ~~whether~~ register spill instructions in spill code generated by a register allocator can be associated with each other;

(b) means for, based on the determining, ~~determining if said register spill instructions can be associated, then~~ rewriting said register spill instructions as a parallel register spill instruction;

(c) means for configuring, based on said rewritten parallel register spill instruction, storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel.

17. (Currently Amended) The system of claim 16, wherein said CPU having parallel registers ~~parallel architecture comprises~~ includes a ~~primary~~ first register set and a ~~secondary~~ second register set, and wherein (a) ~~comprises~~ includes means for determining whether two register spill instructions can be paired.

18. (Currently Amended) The system of claim [[17]] 16, wherein (c) ~~comprises~~ includes means for allocating space on ~~said~~ a memory stack such that paired register spills are double word aligned.

19. (Currently Amended) The system of claim [[18]], further comprising means for loading said paired register spills from said memory stack back into matching register locations in each of said ~~primary~~ first register set and said ~~secondary~~ second register set in parallel.

20. (Currently Amended) A computer readable medium having computer readable program code ~~embedded in the medium~~ for handling register spills in a CPU having parallel registers ~~parallel register architecture~~, the computer readable program code comprising ~~including~~:

(i) code for determining ~~whether~~ that register spill instructions in spill code generated by a register allocator can be associated with each other;

(ii) code for ~~determining if said register spill instructions can be associated, then~~ rewriting, based on the determining, said register spill instructions as a parallel register spill instruction;

(iii) code for configuring, based on said rewritten parallel register spill instruction, storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel.

21. (Currently Amended) The computer useable medium of claim 20, wherein said CPU having parallel registers ~~parallel architecture comprises~~ includes a ~~primary~~ first register set and a ~~secondary~~ second register set, and wherein (i) ~~comprises~~ includes code for determining whether two register spill instructions can be paired.

22. (Currently Amended) The computer useable medium of claim 21, wherein (iii) ~~comprises~~ includes code for allocating space on ~~said~~ a memory stack such that paired register spills are double word aligned.

23. (Currently Amended) The computer useable medium of claim 22, ~~further comprising~~ the computer readable program code further comprising code for loading said paired register

spills from said memory stack back into matching register locations in each of said ~~primary~~ first register set and said ~~secondary~~ second register set in parallel.